CLAIMS

1. Method for deadlock free altering of a network routing from a first routing function $R_{old}$, defining an established connection between a plurality of communication input ports $I_1,..,I_n$ and output ports $O_1,..,O_m$, in a network element, to a second routing function $R_{new}$, defining an new connection between the said input and output ports, for execution by the network element for transmitting and receiving data packets, said method comprising:

(1) for each input port $I_i$, performing the following steps:

(1a) applying the first routing function $R_{old}$ for the input port,

(1b) receiving a token on an input port $I_i$,

(1c) applying the second routing function $R_{new}$ for the input port $I_i$,

(1d) forwarding data packets to every output port $O_j$ associated with the input port $I_i$ according to the second routing function $R_{new}$, provided that the output port $O_j$ has transmitted the token,

(2) for each output port $O_j$, performing the following steps:

(2a) determining if the token has been received on all input ports associated with the output port $O_j$ according to the first routing function $R_{old}$,

(2b) transmitting the token on the output port $O_j$ when the token has been received on all said input ports.

2. Method according to claim 1, wherein the network element is a switch.

3. Method according to claim 1 or 2, wherein the token is included in a data packet.

4. Method according to one of the claims 1-3, wherein the method is applied to deterministic routing functions.

5. Method according to one of the claims 1-4, wherein the method is applied to adaptive routing functions.

6. Method according to one of the claims 1-5, wherein the method is applied to source routing.

7. Method according to claim 5, wherein if the adaptive method gives rise to a cyclic dependency graph, the graph is pruned into a non-cyclic one before the

method is applied.

8. Method according to one of the claims 1-7, wherein the method is applied to only parts of a complete network.

9. Network element, comprising
a plurality of output ports for transmitting data packets to other network elements in a network,
a plurality of input ports for receiving data packets from other network elements in the network,
a processing device,
a memory,
characterized in that the processing device is arranged to perform a method according to one of the claims 1-8.

10. Network element according to claim 9, wherein said routing functions are implemented as tables stored in said memory.

11. Network element according to one of the claims 9 or 10, wherein said memory comprises computer program instructions arranged to perform said method when executed by said processing device.

12. Computer network system, comprising a number of network elements according to claim 9.

13. Computer program, embodied on a storage medium or in a memory, or carried by a propagated signal, for execution by a processing device in a network element,
characterized in that the program comprises a set of instructions arranged to perform a method according to one of the claims 1-8 when executed by the processing device in the network element.